CONTENTS

Chapter No.	TITLE	Page no.
	LIST OF FIGURES	vi
	Title	
1	Title of Topic	1
2	Aim of Study	1
2.1	Introduction	1
2.2	Impact of Malware	1
2.3	Malware detection's significance	3
2.4	Malware Types	4
2.5	Motivation	5
3	Study Background (Literature Review) and significance of study	7
3.1	Literature Survey	7
3.2	Comparative Analysis of existing literature	8
4	Objectives of Study	19
5	National Status	20
6	International Status	20
7	Study Hypotheses	21
8	Study Rationale	21

9	Research Methodology	22
10	Statistical Plan	24
11	Limitations of the Study	25
12	Expected Outcomes of Study	25
13	Bibliography	27

LIST OF FIGURES

Figure No.	Title	Page No.
1.1	Annual number of malware attacks worldwide from 2015 to 2023	3
1.2	Malware Types	5
4.1	Methodology for Malware Detection	22

ABSTRACT

As malware threats become more complex and widespread, there is a growing need for advanced defence systems. This thesis introduces a comprehensive framework designed to improve malware protection in today's rapidly changing threat environment. The framework combines modern detection methods, such as machine learning and deep learning with proactive defence strategies to better identify and respond to malware.

A key feature of the framework is its adaptive learning system, which continuously updates its detection techniques based on new threats and evolving malware tactics. This helps ensure that the system remains effective against both familiar and new types of malware. Additionally, using real-time threat intelligence allows the framework to stay current with the latest threats and quickly adjust its defences.

Implementing this framework involves tackling challenges like integrating advanced technologies and managing significant computational resources. This thesis provides practical advice on how to address these challenges, highlighting the benefits of a flexible and comprehensive approach to malware defence. The proposed framework aims to offer improved protection against the dynamic and growing threat of malware.

1. Title: A Comprehensive Framework for Adapting to New Threats in Malware Defense

2. Aim of Study:

The aim of this study is to enhance malware detection techniques by integrating advanced machine learning and deep learning methods. This research intends to identify and prioritize key factors that contribute to accurate detection, specifically targeting rapidly evolving, polymorphic, and fileless malware. The study also aims to evaluate the effectiveness of deep packet inspection when combined with anomaly and signature-based Intrusion Detection Systems (IDS).

2.1 Introduction:

Malicious software, sometimes known as malware, is a type of malicious code that can cause harm and destruction to digital systems, mobile devices, and computers. The purpose of malware is to steal important data and harm digital systems. A serious danger to a system's privacy, confidentiality, and integrity is malware. Malware comes in many different forms, including viruses, worms, Trojan horses, backdoors, Ransomware, adware, and spyware. One of the largest risks to software applications and other systems, including personal computers, portable laptops, smart devices running the Android operating system, and Internet of Things devices, is the exponential rise in the amount of malware attacks globally, as illustrated in Figure 1.1.

2.2 Impact of Malware:

Malware can have devastating effects on networks, Android devices, and computer systems, compromising security, privacy, and performance.

1. Effect on Networks [3]:

- **Data Breach:** Malware can infiltrate a network to exfiltration sensitive data, leading to severe security breaches. This can include personal information, financial data, intellectual property, and more.
- **Network Disruption:** Malware, such as worms or botnets, can spread across a network, causing congestion, reducing bandwidth, and even bringing down network services through distributed denial-of-service (DDoS) attacks.

- **Unauthorized Access:** Malware can create backdoors or use credentials to allow attackers to remotely control network resources, leading to unauthorized access and potential further exploitation.

2. Effect on Android Devices [2, 4]:

- **Data Theft:** Malware on Android devices can steal personal data, including contacts, messages, and location information. This can be used for identity theft, financial fraud, or spying on the user.
- **Performance Degradation:** Malware can consume system resources, leading to slower performance, increased battery drain, and excessive data usage. This can make the device sluggish and unresponsive.
- **Unwanted Behaviour:** Android malware can display unwanted ads, send premium SMS messages, or download additional malicious apps without the user's consent, leading to financial loss or further compromise.

3. Effect on Computer Systems [5]:

- **System Corruption:** Malware can corrupt files, damage the operating system, or modify critical system settings, leading to system instability or even making the system inoperable.
- **Ransomware:** This type of malware encrypts files on a system and demands a ransom for their release. It can result in the loss of critical data if the ransom is not paid or the decryption key is not obtained.
- **Spyware and Keyloggers:** Malware can monitor user activity, capture keystrokes, and steal sensitive information like passwords and credit card details, leading to severe privacy breaches and financial loss.

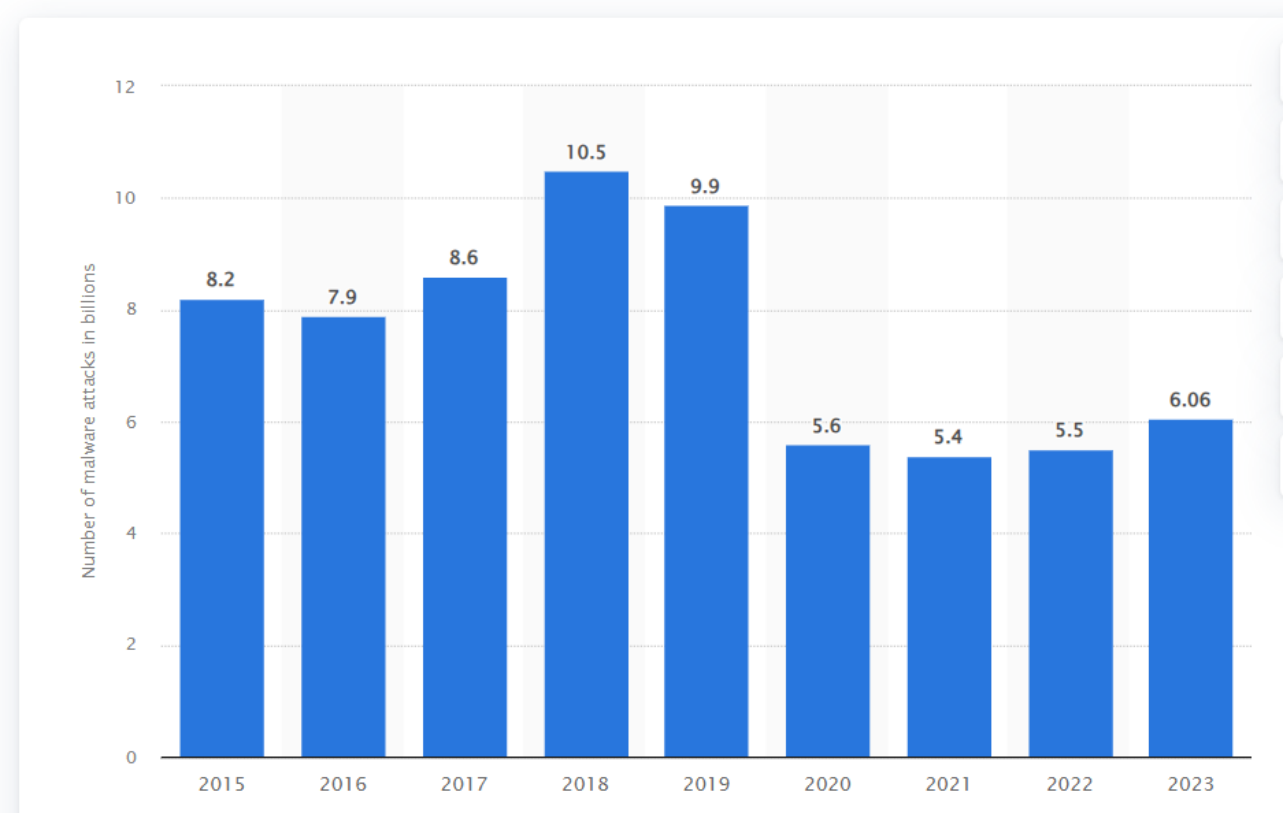

Figure 1.1:- Annual number of malware attacks worldwide from 2015 to 2023 [1]

2.3 Malware detection's significance:

Security: Malware poses significant security threats, including data breaches, theft of sensitive information, and unauthorized access. Effective detection can identify and neutralize these threats before they inflict major harm.

Data Protection: Malware has the potential to steal, corrupt, or destroy data. Detecting malware helps prevent data loss and ensures the integrity and confidentiality of vital information, such as personal, financial, and business-critical data.

System Integrity: Malware can disrupt the normal functioning of computer systems, causing crashes, slow performance, and instability. Detection is crucial for maintaining the stability, reliability, and availability of systems for users.

Financial Impact: Malware can result in financial losses through activities such as fraud, unauthorized transactions, and ransom demands. Early detection helps to minimize these financial risks.

Privacy: Certain types of malware, like spyware and Keyloggers, can invade users' privacy by monitoring activities, capturing sensitive information, and even spying through webcams and microphones. Detection plays a key role in protecting user privacy.

Preventing Propagation: Malware can rapidly spread, infecting multiple systems and devices within a network. Effective detection halts the spread of malware, preventing it from compromising other parts of the infrastructure.

Mitigating Reputation Damage: Malware incidents can severely damage the reputation of individuals, businesses, and organizations. Detecting and mitigating malware helps prevent such damaging events .

Regulatory Compliance: Various industries and regions mandate data protection regulations that require organizations to secure their systems and data. Malware detection is a fundamental component of these compliance efforts .

Preventing Secondary Infections: Malware can create vulnerabilities that other malicious software can exploit. By detecting and removing the initial infection, subsequent attacks can be prevented .

Protection Against Evolving Threats: The malware landscape is constantly changing, with new variants and techniques emerging regularly. Robust detection mechanisms, often enhanced by threat intelligence, are crucial for staying ahead of these evolving threats.

2.4 Malware Types [6]:

Virus: Attaches to files and spreads when those files are executed, often causing damage to data.

Worm: Self-replicates and spreads through networks without needing to attach to other files.

Trojan horse: Disguises itself as legitimate software to perform malicious actions after installation.

Ransomware: Encrypts files and demands a ransom for decryption.

Spyware: Secretly monitors and collects user information without consent.

Adware: Displays unwanted advertisements and can track user activity.

Rootkit: Hides its presence while gaining unauthorized access to a system.

Keylogger: Records keystrokes to capture sensitive information like passwords.

Botnet: A network of infected computers controlled remotely, used for attacks or spam.

Backdoor: Provides unauthorized access to a system, often installed by other malware.

Malvertising: Uses online ads to spread malware through legitimate websites.

Fileless Malware: Operates in memory without creating files on disk, making detection harder

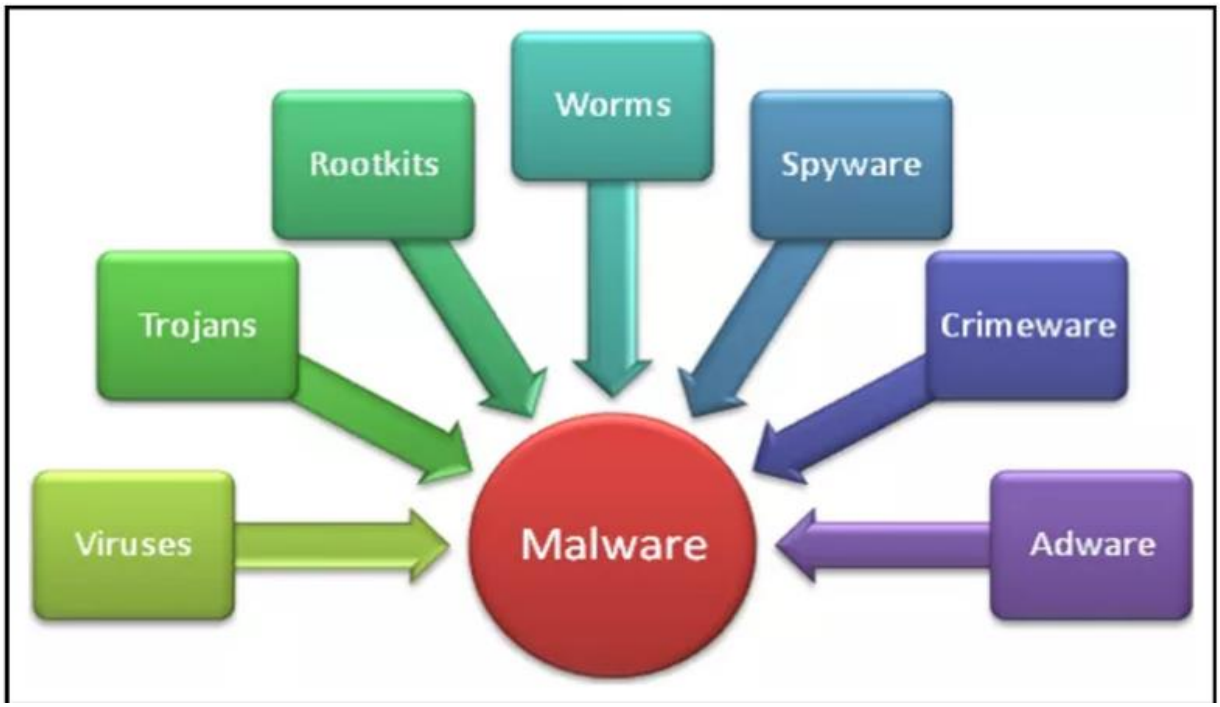

Figure 1.2:- Malware Types [6]

2.5 Motivation:

The motivation for developing advanced malware detection frameworks spans several critical areas. For Intrusion Detection Systems (IDS), the need arises from the ever-evolving threat landscape, where sophisticated and multi-faceted attacks challenge traditional detection methods. Enhanced

frameworks aim to reduce false positives and negatives, improve performance and scalability, and adapt to complex attack patterns. In the realm of Android devices, the growing mobile threat environment necessitates robust detection mechanisms due to the increasing prevalence of malware targeting diverse apps and permissions. These frameworks must also contend with zero-day exploits and evasion techniques while operating efficiently within the resource constraints of mobile devices. For system malware detection, protecting system integrity against threats that may alter critical files, registry settings, or operate file less in memory is crucial. Advanced frameworks must be versatile enough to handle cross-platform threats and ensure user privacy while safeguarding sensitive data. Overall, the development of these frameworks is driven by the need to address dynamic threats, enhance detection accuracy, and provide comprehensive protection across different environments.

3. Study Background (Literature Review) and significance of study

3.1 Literature Survey

The field of cyber security is always changing as new, highly skilled malware strains appear on a regular basis. Research on malware detection is ongoing, so security solutions are always current and effective against the newest threats. Malware is a serious threat to networks, computer systems, and private information. Malware must be found and eliminated in order to stop illegal access, data breaches, and possible harm to software and hardware components. By obtaining private data, including bank account login credentials or credit card details, malware can result in financial losses through identity theft or fraudulent transactions. Malware detection and neutralisation helps shield users from financial loss. Malware can violate users' privacy by tracking their online actions, gathering personal data, and spying on them. Malware detection stops unwanted data collection and helps protect user privacy. Certain malware, like Ransomware, has the ability to encrypt or remove files, resulting in irreversible data loss if the user refuses to pay the ransom. Such data loss can be avoided with early Ransomware detection. Malware has the ability to crash computers, take up system resources, and reduce performance. Malware detection and removal guarantee uninterrupted, effective system operation.

The goal of this research is to create a framework for precisely and quickly identifying harmful software, safeguarding users' devices and systems against online attacks, and enhancing cyber security in general.

3.2 Comparative Analysis of existing literature

In 2022 [1] Statista reported approximately 1.7 billion malware attacks worldwide, marking a significant rise from previous years. This surge underscores the growing threat of cyberattacks and emphasizes the urgent need for improved cyber security measures and defences to tackle the increasing volume and sophistication of malware.

Counter Hack Reloaded [2] by Ed Skoudis and Tom Liston provides a comprehensive guide to understanding computer attacks and developing effective defences. For malware detection, it offers insights into various attack techniques and strategies for identifying and mitigating threats, emphasizing the importance of proactive security measures and detailed knowledge of attack methods to enhance detection and response capabilities.

William Stallings [3] "Network Security Essentials" covers fundamental concepts in network security, including applications and standards crucial for defending against cyber threats. For malware detection, it provides foundational knowledge on network security principles and practices, essential for implementing effective strategies to identify, prevent, and respond to malware attacks.

Eoghan Casey's [4] "Digital Evidence and Computer Crime" offers a thorough exploration of digital forensics and its role in investigating computer crimes. In the context of malware detection, it emphasizes the importance of forensic techniques in identifying, analyzing, and understanding malware, providing insights into how digital evidence can be used to detect and respond to malicious activities effectively.

Gary McGraw's [5] "Software Security: Building Security In" focuses on integrating security practices throughout the software development lifecycle. For malware detection, it highlights the importance of incorporating security measures during development to prevent vulnerabilities that could be exploited by malware, thereby enhancing overall system resilience against malicious threats.

The study by Azeem et al. [6] (2024) analyzes and compares various machine learning approaches for malware detection. It evaluates the effectiveness of different ML techniques in identifying malware, highlighting their strengths and limitations, and provides insights into how these methods can be optimized for improved accuracy and performance in detecting malicious software.

Cui, S., Dong, C.[7] Shen introduces a novel approach using a Transformer-based classifier for malware traffic detection. It addresses the limitations of statistical methods, which struggle with unknown malware traffic and have high false positive rates due to malware variants. The study presents CBSeq's behavior sequence construction and the MSFormer detector, compared against baseline methods like Joy, CIC-Flow, DANTE, and TCC. The approach is validated using the CTU-6 dataset with 328 malware traffic samples.

Djenna, A., Bouridane, A.[8] proposes a novel approach using AI, combining behavior-based deep learning with heuristic methods. This method significantly outperforms traditional static deep learning techniques in malware detection and classification. The study highlights the ineffectiveness of signature-based methods against unknown malware and the challenges posed by advanced evasion techniques. However, it identifies a gap in zero-day malware detection and the

need for proactive security mechanisms. The research utilizes CNN, DNN, random forest, and decision tree classifiers, using the CICAndMal2017 dataset for validation.

Smmarwar, S. K., Gupta [9] provides an in-depth analysis of malware detection methods using machine learning (ML) and deep learning (DL). The review highlights high accuracy in various models but notes challenges with existing feature selection methods, which suffer from low accuracy and high false alarms. It stresses the need for improved feature selection, cross-platform detection techniques, and zero-day malware detection without labeled data. The study references datasets such as CICInvesAndMal2019, CICMalMem2022, and Malimg for validation.

X. Zhang, L. Hao proposes [10] NASP, a neural architecture search method for efficient malware traffic classification (MTC). It optimizes classification performance with fewer parameters and FLOPs, balancing model complexity with performance. However, the study notes the lack of comparison with state-of-the-art NAS methods and limited discussion on scalability. The method is compared with KNN, LR, LeNet, ResNet, CNN, DNAS, and NAS using datasets like USTC-TFC2016 and Edge-IIoTset.

Xu, X., Zhang, X., Zhang [11] introduces the SPCIL method, which enhances malware detection with high accuracy and efficiency. The dual-branch network model balances stability and plasticity, enabling continuous learning of new categories without catastrophic forgetting. The objective function optimizes stability for old classes while maintaining accuracy for new ones. Challenges include traditional methods' difficulty in accurately learning network traffic characteristics and balancing rapid processing with memory retention. The study uses the USTC-TFC2016 dataset with 142,700 samples across 20 categories.

Rao, S. M., & Jain A. [12] highlights how cloud computing enhances malware detection by overcoming traditional limitations. The review discusses a hybrid model that improves malware categorization using deep learning and introduces an innovative security design for monitoring virtual machines to detect attacks. It also addresses challenges such as the complexity of traditional models, difficulty in detecting novel malware, and issues with unknown, imbalanced, and scarce data. The study uses malware logs from 51 families and 17,394 data points from various locations for analysis.

Zhang, W., & Lazaro, J. P. [13] reviews methods for detecting network anomalies using statistical models, machine learning (ML), and deep learning (DL). It discusses trends, challenges, and future

developments, highlighting issues with statistical models' reliance on data distribution assumptions and the high resource consumption of behavior analysis methods. The paper also notes the lack of unified standards for behavior modeling and real-time performance in anomaly detection. Techniques reviewed include clustering, graph analysis, and Markov chain methods, with the OpenVC and KDD-CUP99 data sets used for model evaluation.

Aslan, O., & Samet, R.[14] examines various detection methods, challenges, and new assumptions. It introduces SCTPL for detecting malicious behaviors and proposes a model-checking algorithm. The review highlights the limitations of current methods, such as false positives and negatives, bias, overfitting, and the inability to detect all sophisticated malware due to complexity and obfuscation. The paper covers signature-based, behavior-based, model-checking, cloud-based, and deep learning-based detection methods, utilizing datasets like NSL-KDD, Drebin, Microsoft Malware Classification, ClaMP, AAGM, and EMBER for evaluation.

Zheng, X., & Li, H.[15] The paper focuses on identifying malicious encrypted traffic using deep learning, specifically through a proposed MET-FMF scheme based on multi-feature fusion. It highlights the challenges of data imbalance in real-world datasets, which can lead to model overfitting. The study uses the ISCXVPN2016 dataset, split into a 70:30 ratio for training and testing, and compares the proposed method with other schemes like MARK-ELM, VAEN, and HAST-IDS. The research also addresses the need for improved sampling techniques to handle data imbalance and emphasizes the importance of data preprocessing and model selection.

Sihag [16] enhances Android malware detection using network traffic packet inspection, achieving 99.12% accuracy. However, it lacks comparisons with real-time detection systems and offers limited discussion on scalability. The approach involves dynamic analysis of application behavior and converting packet inspection data into images for CNN training.

Merve Çelebi, Alper Özbilen, Uraz Yavanoğlu reviews [17] Deep Packet Inspection (DPI) challenges in advanced network traffic analysis, focusing on IoT and SDN security. It discusses DPI's application to encrypted traffic and identifies performance limitations due to cryptographic calculations. The paper outlines implementation challenges, proposes a future roadmap for DPI techniques, and uses open-source DPI modules with real datasets for evaluation.

Reddy, P. C. S., Shirley Muller, P., N Koka, S., Sharma, V., Sharma, N., & Mukherjee, S.[18] presents the HexCNN-1D model for classifying malicious traffic. Key limitations include low accuracy due to unequal data distribution and the need for normalized processing. The research

highlights the lack of focus on multi classification in previous work and limited exploration of interior covariate shifts. The model incorporates Global-Attention-Block (GAB) and Category-Attention-Block (CAB) modules, using the ISCX-VPN-NonVPN-2016 and USTC-TFC2016 datasets for evaluation.

DroidMD [19] is an Android malware detection tool that achieves 95.5% accuracy at the source code level, analyzing 30,000 apps and detecting 3,670 malware apps. However, it struggles with distinguishing malware clone fragments from original code, is limited to mobile markets, and cannot detect malware on devices or in other programming languages. The study uses 27,000 benign apps from AppChina and 3,670 malware apps from sources like Genome and MalShare.

Maniriho, P., Mahmood, A. N. proposes [20] a Windows malware detection approach using API call sequences with transformer models. It enables early detection but is limited by a small dataset size and potential biases that may generate irrelevant sequences. The study employs generative transformer models, deep recurrent neural networks, and bidirectional GRU with attention mechanisms. The model is trained on API call datasets from various sources.

Zhao, X., Zhang, R., & Guo, S. [21] present the "ERNN" model for encrypted traffic detection using the IDS2017 and IDS2018 datasets, focusing on multi-classification to identify specific types of attacks. The study evaluates performance with accuracy and F1 metrics but provides limited discussion on real-world applications and lacks comparisons with real-time intrusion detection systems. The model incorporates a session gate with four action types and Mealy machine updates.

Galli, A., la Gatta, V., Moscato, V., Postiglione [22] explores the application of Explainable AI (XAI) techniques to LSTM models used for detecting malware. It emphasizes the importance of explanation efficiency and examines various explainer modules, such as LIME, SHAP, LRP, and attention mechanisms, to enhance the transparency of these models. Despite the growing use of deep learning in malware detection, the paper identifies a significant gap in the research on explainability within the behavioral malware detection domain, particularly in the analysis of sequential behavior rather than static features. Additionally, it highlights challenges such as the lack of qualitative metrics for evaluating XAI and domain-specific limitations that impact the effectiveness of explanations. The study utilizes datasets like Mal-API-2019, API Call Sequences, and Alibaba Cloud Malware to demonstrate the practical application of these techniques, underlining the need for improved XAI methods to build trust and understanding in AI-driven security systems.

Bader, O., Lichy, A., Hajaj, C., Dubin, R.[23] introduces the MalDIST model, which outperforms both machine learning and deep learning approaches in classifying malware traffic. MalDIST effectively transfers encrypted traffic classification techniques to malware detection, achieving high accuracy, precision, and recall. However, the study highlights the need for evaluation on industry-level datasets to account for network variability and malware diversity. Additionally, it calls for more diverse datasets to better validate deep learning architectures, focusing on detection complexity and speed. The research utilizes datasets such as StratosphereIPS, ISCX2016, and Malware-Traffic-Analysis.net for testing.

Weitkamp, E., Satani, Y., Omundsen [24] showcases the use of machine learning, particularly an LSTM model, to detect IoT malware traffic with higher accuracy than SVM. It leverages Apache Kafka and Spark for real-time detection with a scalable, enriched dataset. The study notes that signature-based methods are less effective against unknown attacks, and while LSTM outperforms SVM, there's a lack of comparison with traditional cybersecurity methods. The research uses the Aposemat IoT-23 and ToN IoT datasets for evaluation.

Naseer, M., Rusdi, J. F., Shanono [25] examines current malware detection techniques, focusing on enhancing detection accuracy and reducing false positives and negatives. It stresses the need for improved feature selection algorithms and highlights challenges like limited computing resources and inadequate evaluation of malware samples. The study also discusses the importance of optimizing signature repositories for IoT devices and uses datasets like APK Auditor client and runtime system calls.

Liu, J., Wang, L., Hu, W., Gao [26] presents the TLARNN model, which uses spatial-temporal features and a dual-attention mechanism to detect encrypted malicious traffic, outperforming existing methods. However, it highlights the need for better sampling methods and improved consideration of forward and backward data packet correlations. The study employs the CICAnd-Mal2017 dataset for evaluation and compares TLARNN with rule-based detection algorithms like DPI-based methods.

Zhu, H. J., Wang, L. M. Zhong [27] introduces HDNFDroid, which enhances malware detection using Merged Sparse Auto-Encoder (MSAE) and Stacked Hybrid Learning MSAE and SDAE (SHLMD). The framework improves feature representation, outperforming existing methods. However, classical methods like Isomap and t-SNE underperformed compared to the baseline

RawF. The study uses the MUDFLOW dataset and a self-collected dataset for evaluation, employing SVM and KNN for classification.

Feng, R., Chen, S., Xie, X., Meng [28] focuses on using deep learning (DL) for malware detection on mobile devices, utilizing RNN, LSTM, GRU, and CNN architectures within the MobiTive system to ensure performance and efficiency. It highlights limitations in feature categories for detecting new malware families and suggests combining dynamic and static methods for improved detection. Feature selection is based on manifest properties and API calls, with behavior-based updates enhancing feature representation. The study is conducted on a dataset of 70,000 Android apps, including 29,010 malware and benign samples, with 21 virus labels.

Dong, S., Shu, L., & Nie, S introduce [29] D-CNN method boosts malware detection accuracy to 96.80%, addressing the limitations of traditional methods like low accuracy and high false alarm rates. Despite its success, there's limited research on Android exploit permissions, API call graphs, and other hybrid deep learning structures. The study uses Min-Max normalization for data preprocessing and evaluates the model on the Drebin dataset, which includes 5,560 malicious and 5,560 benign samples from the Google Play Store.

Shu, L., Dong, S., Su, H., & Huang [30] uses opcode sequences without feature engineering, evaluated by accuracy, F1 score, and NLP techniques. However, it struggles with encrypted or obfuscated malware, and concept drift reduces early detection accuracy. The study highlights a need for hybrid models and interpretable CNN algorithms, combining static, dynamic, and hybrid analyses with NLP, CNNs, LSTM, and attention mechanisms. Datasets include Androzoo, Google Play Store, Drebin, and VirusTotal.

Kim, C., Chang, S. Y., Kim, J., Lee [31] focuses on zero-day malware detection using autoencoding and one-class (OC) classification, combining these techniques for robust detection. While OC classifiers underperform compared to supervised methods and autoencoding requires precise threshold settings, the research proposes an effective methodology with high-quality models. The study lacks comparison with traditional unsupervised methods and exploration of varying training set sizes. Evaluation was conducted using the Meraz'18 and Drebin datasets.

Fang, W., He, J., Li, W., Lan, X., Chen [32] enhances Android malware detection through federated learning and genetic evolution, achieving a 98.53% F1 score, outperforming baseline classifiers and state-of-the-art methods. However, the approach faces issues like gradient disappearance and

increased computational overhead in federated training. The study lacks comparison with newer federated learning architectures and has limited discussion on scalability. It uses genetic evolution for malware simulation, a residual neural network for accuracy, and multitask learning for anomaly detection in IoT devices. The CIC, Contagio, and Drebin datasets were used for evaluation.

Li, D., Cui, S., Li, Y., Xu, J [33] proposed PAD framework significantly enhances malware detection against evasion attacks, achieving high accuracy and outperforming existing defenses. However, it lacks theoretical guarantees in heuristic methods and struggles to analyze specific attack types thwarted by defense models. The approach uses gradient-based methods, convex measurements for perturbation quantification, and direct mapping of perturbation vectors to the problem space.

Habib, F., Shirazi, S. H., Aurangzeb [34] detecting metamorphic malware in IoT devices, with EfficientNetB0 outperforming other models. The Malimg dataset is used for classification, transforming labels into numerical formats for machine learning. While the study effectively addresses dynamic code challenges in IoT security, it lacks exploration of real-world implementation and scalability issues. Evaluation metrics include F1-score, precision, recall, and accuracy, with comparisons made across models like ResNet50, MobileNet, and VGG16.

Redhu, A., Choudhary, P., Srinivasan [35] significantly enhanced malware detection in cyberspace, offering high accuracy and adaptability to emerging threats. However, challenges like class imbalance, model interpretability, and detecting obfuscated malware remain critical. Signature-based methods struggle with polymorphic malware, while behavior-based detection faces complexity. Datasets like KDD '99, NSL-KDD, and IoT-23 are used for evaluation, with architectures like LSTM and Extreme Learning Machines improving detection performance.

Kumar [36] explores scalable malware detection using a distributed deep learning approach, focusing on accuracy, scalability, and efficiency. A CNN-BiLSTM model is utilized for high accuracy, addressing challenges like novel malware threats and obfuscated code. However, there's limited discussion on real-time detection and scalability in distributed deep learning. The study uses datasets like VirusShare, VirusTotal, and Contagio for evaluation.

Schmitt, M [37] examines AI-based cyber threat detection in digital ecosystems, highlighting integration challenges and the potential for AI misclassification. It emphasizes the need for stronger AI classification and research on AI-human collaboration. The study reviews literature on AI/ML in

cybersecurity and experiments with various models, including logistic regression, random forest, and deep learning. Datasets used include Network Intrusion Detection (25,192 observations), Android Malware Detection (15,036 observations), and IoT Cyber Threat Detection (157,800 observations).

Nobakht, M., Javidan, R., & Pourebrahimi [38] employ ensemble learning with 1D-CNNs and Random Forest for accurate IoT malware detection. It faces challenges with model complexity, time consumption, and IoT device constraints. The study lacks comparisons with real-world scenarios and exploration of scalability. The model uses a stack of 1D-CNNs and Random Forest for final predictions, with hyperparameter optimization. Datasets include IoT-23 and Stratosphere IPS.

Bhardwaj, S., & Dave [39] uses a CNN-1D model to enhance attack detection with plans for automated node scoring and real-time identification. Challenges include privacy restrictions on packet capture files and limited training data affecting accuracy. The study lacks real-time prediction capabilities and requires optimization for complex network graphs. The approach involves packet-level and graphical analysis, with datasets like CSE-CIC-IDS2018, UNSW-NB15, and CIC-Darknet2020.

Qin, M., Zhang, X., Wang, praposes [40] NAS-based FS-MTC method that achieves high accuracy in malware classification. Challenges include limited samples for mapping and reliance on ensemble classifiers. The study lacks comparisons with traditional methods and exploration of extreme one-shot tasks in IoT security. It utilizes KNN, SVM, and logistic regression models, with the USTC-TFC2016 and CIC-IoT datasets.

Fang, W., He, J., Li, W [41] utilizes federated learning and genetic evolution for Android malware detection, achieving high accuracy, especially in cross-dataset tests. Challenges include low detection rates due to malware evolution limitations and privacy concerns with centralized data. It employs a genetic evolution strategy and federated learning framework, with datasets including CIC, Contagio, and Drebin.

Amer, E., Samir, A. [42] uses API call sequences and Ant-Colony Optimization (ACO) to distinguish between malicious and non-malicious apps. It employs TF-IDF and Word2Vec for statistical and contextual weighting, with a swarm intelligence algorithm creating a behavioral

matrix. Challenges include static analysis limitations and antivirus signature databases' lag. The method's effectiveness is tested with the Brazilian-Malware dataset and a Good ware dataset.

Pan, Z., Sheldon [43] leverages techniques like explainable machine learning to enhance accuracy and transparency. Approaches include using linear regression to improve detection efficiency, RNNs for handling sequential data, and importance ranking through regression coefficients. However, there are challenges such as reliability issues in HPC-based methods, insufficient validation, and a lack of explainability in existing hardware-based approaches. Tools and techniques employed include Hardware Performance Counters (HPC), Embedded Trace Buffers (ETB), and a comparison of Decision Trees (DT) and LSTM-based RNNs, often using a combination of HPC and ETB data sources.

Hashmi, S. A. proposed [44] Android malware detection benefits from ECNN models, which offer high accuracy and enhance cyber security by identifying and categorizing malware. Traditional antivirus methods struggle with evolving threats, and static analysis is hindered by dynamic code loading and obfuscation, while dynamic analysis is costly and risky. Attackers also use anti-emulation tactics to avoid detection. Detection performance gaps for new malware and insufficient study of key characteristics for Android malware differentiation are ongoing issues. Tools and techniques include Enhanced Convolutional Neural Networks (ECNN), Support Vector Machines (SVM) with SGD optimization, and weighted features for SVM learning. Datasets used include Android Malware Dataset-1 (user behaviour), Dataset-2 (accuracy), and Dataset-3 (recall evaluation).

The literature review highlights several research gaps and challenges in detecting Malware

- **Evolving Malware Threats:**
    - Rapidly evolving malware techniques make it difficult to keep detection methods up-to-date.
    - Zero-day attacks and new variants often bypass traditional detection systems.
- **Obfuscation and Encryption:**
    - Malware often uses obfuscation and encryption to hide its presence and activities.
    - Detecting malicious features within encrypted traffic poses significant challenges.
- **High False Positives and Negatives:**

- Balancing false positives (FPs) and false negatives (FNs) is difficult, leading to either missed threats or excessive alerts.
- Close similarities between benign and malicious activities contribute to detection errors.

- **Data Imbalance:**
  - Real-world data is often imbalanced, with fewer samples of malware compared to benign data.
  - This imbalance can lead to over fitting and poor generalization in detection models.
- **Scalability Issues:**
  - Ensuring detection methods scale effectively in high-traffic and large-scale network environments is challenging.
  - Real-time performance requirements add to the complexity.
- **Complex and Diverse Environments:**
  - Malware detection must be effective across various platforms and network environments, including IoT, cloud, and mobile devices.
  - Varying network traffic patterns and behaviours require adaptable detection methods.
- **Feature Selection and Optimization:**
  - Identifying the most relevant features for accurate malware detection is complex.
  - Optimal feature selection is crucial for improving detection rates and reducing false alerts.
- **Proactive and Adaptive Security:**
  - Implementing proactive security mechanisms that adapt to evolving threats is challenging.
  - Balancing the need for dynamic detection with resource constraints is difficult.
- **Real-time Analysis and Response:**
  - Achieving real-time data extraction, analysis, and response to threats requires high-performance systems.
  - Latency and resource consumption can impede real-time capabilities.
- **Ethical and Legal Considerations:**
  - Ensuring compliance with data privacy regulations and ethical standards in detection practices.

    - Addressing legal implications of monitoring and analyzing network traffic for security purposes.
- **Integration with Advanced Technologies:**
    - Effectively integrating deep learning and hybrid models with existing detection frameworks.
    - Ensuring new technologies enhance detection without adding excessive complexity.
- **Behavioural Analysis Limitations:**
    - Accurately modelling and analyzing malware behaviour to detect sophisticated threats.
    - Differentiating between benign and malicious behaviours, especially in complex environments.
- **Cross-platform Detection:**
    - Developing detection methods that work effectively across different operating systems and hardware platforms.
    - Ensuring consistent performance and accuracy in diverse technological ecosystems.

4. Objectives of Study

The research contributes to developing and evaluating innovative frameworks to address the challenges in malware detection, we propose a framework that integrates advanced techniques across IDS, system, and Android environments. Current methods struggle with unknown malware traffic, high false positive rates, and evasion tactics used by modern malware. Signature-based methods fail to detect novel threats, while existing feature selection and behaviour analysis techniques suffer from low accuracy and high resource consumption. Our framework aims to overcome these limitations by combining anomaly-based and signature-based IDS with deep packet inspection, leveraging machine learning and deep learning technologies to enhance detection. It will address issues with data imbalance, feature dimensionality, and encrypted or obfuscated malware. Additionally, the framework will incorporate solutions for IoT and Android-specific challenges, including dataset constraints and privacy concerns, to improve real-time detection and adaptability to evolving threats.

This research will be conducted focusing following objectives:

1. To identify and prioritize key factors for accurate malware detection by applying advanced machine learning and deep learning techniques.
2. To propose a method for detecting and managing, polymorphic malware that evade traditional detection techniques.
3. To develop a framework that combines deep packet inspection with anomaly and signature-based IDS, and learns attack behaviour to improve malware detection.

5. National Status

In recent years, national efforts in malware detection have focused on developing AI-based solutions to protect critical infrastructures and digital industries. Researchers in the country have been exploring various machine learning models and deep learning architectures to detect malware in different environments, including mobile devices and IoT networks (Jabeen et al., 2019; Maniriho et al., 2024). National research institutions have also been actively collaborating with industry partners to develop scalable malware detection systems using distributed deep learning techniques (Kumar, 2023).

6. International Status

Internationally, there has been significant progress in malware detection, driven by the increasing adoption of advanced AI models. Global studies have explored the effectiveness of different machine learning and deep learning approaches, including federated learning and self-paced class incremental learning, to detect malicious traffic (Xu et al., 2024; Bhardwaj & Dave, 2023). Researchers have also been developing sophisticated models to analyze encrypted traffic and apply hybrid detection methods in cloud computing environments (Rao & Jain, 2024; Zheng & Li, 2023). Furthermore, international collaborations have been enhancing AI-based behavioural malware detection systems' transparency and explain ability (Galli et al., 2024).

Some notable trends in international malware detection include:

Increased Focus on Advanced Persistent Threats (APTs): These sophisticated attacks often target critical infrastructure and government agencies, necessitating advanced detection techniques.

Rise of Ransomware: The prevalence of Ransomware attacks has led to increased efforts to prevent and mitigate their impact.

Development of New Detection Technologies: Machine learning, artificial intelligence, and behavioural analytics are being employed to identify and respond to emerging malware threats.

Challenges in Developing Nations: Countries with limited resources and infrastructure may face particular difficulties in implementing effective malware detection measures.

7. Study Hypotheses:

H1: Integrating deep learning techniques with deep packet inspection improves the detection accuracy of encrypted and polymorphic malware compared to traditional intrusion detection methods.

This hypothesis combines the key elements of previous hypotheses, focusing on the effectiveness of deep learning and deep packet inspection in detecting advanced malware types, such as encrypted, polymorphic, and fileless malware. It also implies a comparative analysis against traditional methods, making it more comprehensive and aligned with current research trends.

H0: Integrating deep learning techniques with deep packet inspection does not significantly improve the detection accuracy of encrypted and polymorphic malware compared to traditional intrusion detection methods, due to the complexity of analyzing encrypted traffic and the challenges in training models on diverse and evolving malware signatures.

This hypothesis suggests that despite the advanced capabilities of deep learning, the effectiveness of malware detection might be limited by the inherent complexities of encrypted data, the need for substantial computational resources, and the potential for high false positive rates in detecting rapidly evolving malware types.

8. Study Rationale:

The rapid evolution of malware, characterized by an increasing number of attacks and the growing sophistication of attack vectors, has created significant challenges for cyber security. Traditional malware detection methods, such as signature-based and rule-based approaches, often fail to detect new, unknown, or rapidly mutating malware variants, including polymorphic and fileless malware. These traditional methods are limited by their reliance on predefined signatures or static rules, which are easily evaded by advanced malware that can change its code, behaviour, or conceal itself within legitimate processes.

Recent trends indicate a substantial rise in malware attacks, both in volume and complexity, with encrypted and network-based threats becoming more prevalent. As malware continues to adapt and evolve, there is a critical need for innovative detection methodologies that can handle these emerging threats effectively. This study aims to fill this gap by leveraging advanced machine learning (ML) and deep learning (DL) techniques, integrated with intrusion detection system (IDS) and deep packet inspection (DPI). ML and DL offer the potential to learn from large datasets, recognize patterns, and detect anomalies, even in encrypted or obfuscated network traffic.

By integrating these techniques with DPI, the study proposes a novel framework capable of analyzing data at both the network level and the application layer. This approach can identify malicious behaviours more accurately and in real-time, even when dealing with sophisticated malware that evades traditional detection methods. Furthermore, the study seeks to enhance detection rates, reduce false positives, and provide a scalable solution that adapts to new malware types as they emerge. This research addresses a critical gap in cyber security and aims to advance the field by developing a more effective and resilient malware detection framework that aligns with the current and future landscape of cyber threats.

9. Research Methodology: This research will be carried out following the methodology presented in figure 4.1.following stages will be carried out. Methodology for Malware Detection Framework Integrating IDS, DPI, System, and Android.

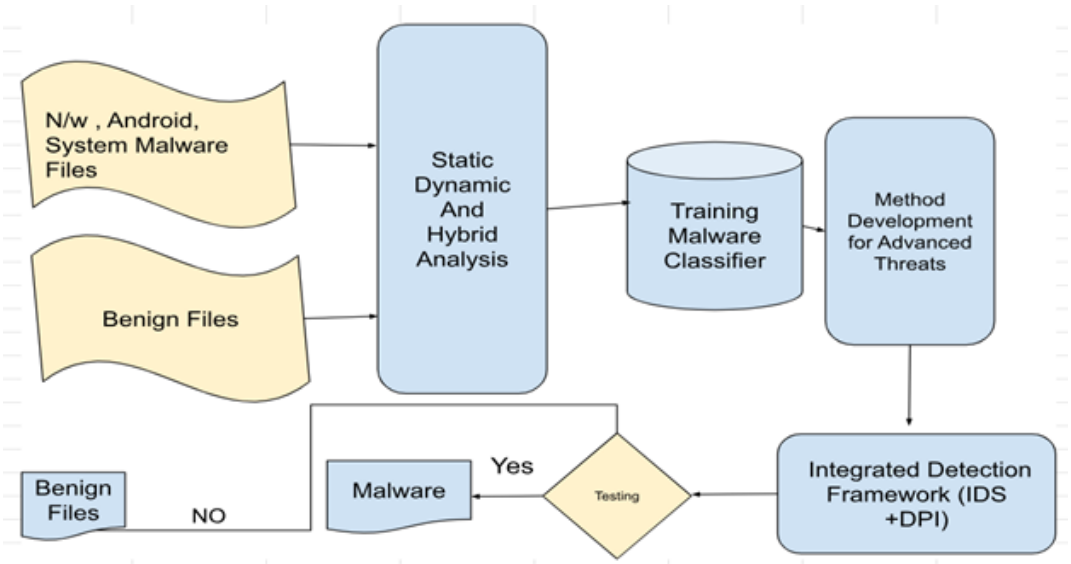

Figure 9.1:- Methodology for Malware Detection

Input Data and Pre-processing: To address the challenge of robust malware detection amidst variant malware and real-world conditions, the framework starts with comprehensive data collection from diverse sources: network traffic, system logs, and Android application data. This data undergoes rigorous pre-processing to remove noise and standardize it for consistency. Techniques such as Min-Max normalization are applied to scale data effectively, while advanced oversampling methods tackle the issue of data imbalance that often leads to over fitting. Dynamic

analysis for system and Android applications ensures that data handling is optimized, and real-time data acquisition methods are employed to streamline processing and adapt to evolving malware threats.

Feature Extraction: Feature extraction is critical for overcoming the limitations of existing methods that struggle with zero-day malware and real-time detection. By leveraging advanced behaviour sequence construction techniques, such as CBSeq, and incorporating transformer-based classifiers and deep learning models (e.g., CNNs, LSTMs), the framework captures and represents malware traffic patterns with high accuracy. This includes employing sophisticated feature selection techniques and dimensionality reduction methods to refine feature quality and mitigate the complexities introduced by malware obfuscation and evolving variants. Techniques like TF-IDF and Word2Vec are used for contextual and statistical weighting, enhancing feature relevance and detection precision.

Method Development for Advanced Threats: During process fitting, the focus is on developing and optimizing models to handle diverse and unknown malware effectively. This phase integrates both machine learning (ML) and deep learning (DL) approaches:

- **Machine Learning Approaches:** Classical ML models like Random Forest, Decision Trees, and Support Vector Machines (SVMs) are employed for their ability to provide interpretability and manage different types of features. These models are trained to detect malware based on the features extracted during the previous phase. Ensemble methods, such as boosting and bagging, enhance these traditional ML models by combining multiple classifiers to improve detection accuracy and robustness.
- **Deep Learning Approaches:** Deep learning models, including Convolutional Neural Networks (CNNs), Bidirectional Long Short-Term Memory Networks (BiLSTMs), and custom architectures like HexCNN-1D, are utilized for their superior ability to handle complex and high-dimensional data. CNNs are particularly effective in extracting spatial hierarchies in data, while BiLSTMs are adept at capturing temporal dependencies. Hybrid models combining CNNs and LSTMs provide a comprehensive approach to sequence and pattern recognition. Advanced techniques such as Neural Architecture Search (NAS) and hyper parameter optimization are employed to tailor the models to specific detection tasks and improve their performance.

Malware Detection Using integrated Framework: For robust malware detection, the framework employs an ensemble approach that combines IDS, DPI, system, and Android data. This integration allows for the effective handling of diverse malware threats through various detection methods, including DPI-based traffic classification, IDS for anomaly detection, and deep learning models for behaviour analysis. Techniques like SPCIL and NASP-MTC address the need for stability and adaptability in the detection framework, handling both known and unknown malware effectively. The ensemble approach leverages the strengths of different models, providing comprehensive coverage and resilience against evolving malware threats. Additionally, the framework incorporates real-time performance metrics and scalability considerations to ensure effective adaptation to new and unknown malware variants.

Testing: To test the malware detection framework, a comprehensive evaluation will be conducted on its integrated components, including IDS, DPI, and system and Android data, using a broad dataset of known and unknown malware types. Real-time detection capabilities will be assessed by measuring latency, throughput, and detection rates under dynamic conditions. Stability and adaptability will be tested with techniques like SPCIL and NASP-MTC to ensure the framework effectively handles new malware variants. Performance metrics and scalability will also be evaluated across varying dataset sizes and network loads to confirm robust, real-time threat detection.

10. Statistical Plan:

Sr. No.	Task	Year 1 (2024-25)	Year 2 (2025-26)	Year 3 (2026-27)
1	Literature Review & Research Design	Months 1-6		
2	Data Collection & Initial Analysis	Months 7-12		
3	Conference Paper 1: Preparation & Submission	Months 7-12		
4	Journal Paper 1: Drafting & Submission		Months 13-16	
5	Further Research & Data Analysis		Months 17-22	
6	Conference Paper 2: Preparation & Submission		Months 23-28	
7	Journal Paper 2: Drafting & Submission		Months 23-28	
8	Thesis Writing & Final Revisions			Months 29-36

11. Limitations of the Study:

1. The study relies on publicly available datasets, which may not fully represent real-world network environments.
2. The effectiveness of the proposed framework may vary depending on network configurations and traffic patterns.
3. The study may face challenges in generalizing results due to the dynamic nature of malware evolution
4. Limited availability of real-world malware datasets may impact the generalizability of the findings.
5. The study may face challenges in detecting malware variants that use highly advanced evasion techniques.
6. Potential bias in dataset labelling could affect the training and evaluation of machine learning models.

12. Expected Outcomes of Study:

This research aims to address the critical objectives for advancing malware detection by proposing a framework that demonstrates a robust and comprehensive approach by integrating advanced machine learning (ML) and deep learning (DL) techniques. This framework effectively identifies and prioritizes key factors crucial for accurate malware detection through the following strategies:

1. **Advanced Techniques for Accurate Detection:** By utilizing cutting-edge machine learning models and deep learning architectures, the framework can effectively analyze large and complex datasets. This approach enhances detection accuracy and addresses the diverse and evolving nature of malware threats.
2. **Detection of Evasive Malware:** The framework addresses the challenge of detecting rapidly changing, polymorphic, and files less malware that often evades traditional detection methods. Through sophisticated feature extraction techniques, behaviour sequence analysis, and hybrid DL models, it effectively captures and responds to malware that adapts to evade detection.
3. **Integration of DPI and IDS:** By combining deep packet inspection (DPI) with anomaly-based and signature-based IDS, the framework enhances its ability to detect and manage

malware across various contexts. This integration leverages the strengths of each method, improving overall detection capabilities and resilience against diverse threats.

4. **Behaviour-Based Detection Framework:** The framework incorporates methods that learn and analyse attack behaviours, enabling it to detect malware based on behavioural patterns rather than solely relying on static signatures. This dynamic approach allows for more accurate detection of new and evolving threats.
5. **Real-Time Performance Evaluation:** Emphasizing the need for immediate threat detection, the framework is designed with real-time performance metrics in mind. By incorporating distributed architectures and real-time data processing techniques, it ensures prompt identification and mitigation of malware attacks, minimizing potential damage and enhancing overall security responsiveness.

13. BIBLIOGRAPHY

1. Number of malware attacks per year 2022 Statista. Statista (2023). https://www.statista.com/statistics/873097/malware-attacks-per-year-worldwide/
2. Skoudis, Ed, and Tom Liston. Counter Hack Reloaded: A Step-by-Step Guide to Computer Attacks and Effective Defense. Prentice Hall, 2006.
3. Stallings, William. Network Security Essentials: Applications and Standards. Pearson, 2017.
4. Casey, Eoghan. Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet. Academic Press, 2011.
5. McGraw, Gary. Software Security: Building Security In. Addison-Wesley, 2006.
6. Azeem, Muhammad, et al. "Analyzing and comparing the effectiveness of malware detection: A study of machine learning approaches." Heliyon 10.1 (2024).
7. Cui, Susu, et al. "CBSeq: A Channel-level Behavior Sequence For Encrypted Malware Traffic Detection." IEEE Transactions on Information Forensics and Security (2023).
8. Djenna, Amir, et al. "Artificial intelligence-based malware detection, analysis, and mitigation." Symmetry 15.3 (2023): 677.
9. Smmarwar, Santosh K., Govind P. Gupta, and Sanjay Kumar. "Android Malware Detection and Identification Frameworks by Leveraging the Machine and Deep Learning Techniques: A Comprehensive Review." Telematics and Informatics Reports (2024): 100130.
10. Zhang, Xixi, et al. "An automatic and efficient malware traffic classification method for secure Internet of Things." IEEE Internet of Things Journal (2023).
11. Xu, Xiaohu, et al. "Advancing Malware Detection in Network Traffic with Self-Paced Class Incremental Learning." IEEE Internet of Things Journal (2024).
12. Rao, S. Madhusudhana, and Arpit Jain. "Advances in Malware Analysis and Detection in Cloud Computing Environments: A Review." International Journal of Safety & Security Engineering 14.1 (2024).
13. Zhang, Weibao, and Joan P. Lazaro. "A Survey on Network Security Traffic Analysis and Anomaly Detection Techniques." International Journal of Emerging Technologies and Advanced Applications 1.4 (2024): 8-16. https://www.ijetaa.com/article/view/117/

14. Aslan, Ömer Aslan, and Refik Samet. "A comprehensive review on malware detection approaches." IEEE access 8 (2020): 6249-6271. https://doi.org/10.1109/ACCESS.2019.2963724.
15. Zheng, Xianchun, and Hui Li. "Identification of malicious encrypted traffic through feature fusion." IEEE Access 11 (2023): 80072-80080. https://doi.org/10.1109/ACCESS.2023.3279120.
16. Sihag, Vikas, et al. "PICAndro: Packet InspeCtion-Based Android Malware Detection." Security and Communication Networks 2021.1 (2021): 9099476. https://doi.org/10.1155/2021/9099476
17. Çelebi, Merve, Alper Özbilen, and Uraz Yavanoğlu. "A comprehensive survey on deep packet inspection for advanced network traffic analysis: issues and challenges." Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi 12.1 (2023): 1-29 https://doi.org/10.28948/ngmuh.1184020
18. Reddy, Pundru Chandra Shaker, et al. "Detection of Encrypted and Malicious Network Traffic using Deep Learning." 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE). IEEE, 2023. https://doi.org/10.1109/AIKIIE60097.2023.10390386
19. Akram, Junaid, et al. "DroidMD: an efficient and scalable android malware detection approach at source code level." International Journal of Information and Computer Security 15.2-3 (2021): 299-321. https://doi.org/10.1504/ijics.2019.10020453
20. Maniriho, Pascal, Abdun Naser Mahmood, and Mohammad Jabed Morshed Chowdhury. "EarlyMalDetect: A Novel Approach for Early Windows Malware Detection Based on Sequences of API Calls." arXiv preprint arXiv: 2407.13355 (2024). http://arxiv.org/abs/2407.13355
21. Zhao, Ziming, et al. "ERNN: Error-resilient RNN for encrypted traffic detection towards network-induced phenomena." IEEE Transactions on Dependable and Secure Computing (2023). https://doi.org/10.1109/TDSC.2023.3242134
22. Galli, Antonio, et al. "Explainability in AI-based behavioral malware detection systems." Computers & Security 141 (2024): 103842. https://doi.org/10.1016/j.cose.2024.103842
23. Bader, Ofek, et al. "MalDIST: From encrypted traffic classification to malware traffic detection and classification." 2022 IEEE 19th annual consumer communications &

networking conference (CCNC). IEEE, 2022. https://doi.org/10.1109/CCNC49033.2022.9700625

24. Weitkamp, Ethan, et al. "MalIoT: Scalable and Real-time Malware Traffic Detection for IoT Networks." arXiv preprint arXiv: 2304.00623 (2023).. http://arxiv.org/abs/2304.00623
25. Naseer, Muchammad, et al. "Malware detection: issues and challenges." Journal of Physics: Conference Series. Vol. 1807. No. 1. IOP Publishing, 2021.https://doi.org/10.1088/1742-6596/1807/1/012011
26. Liu, Jianyi, et al. "Spatial-Temporal Feature with Dual-Attention Mechanism for Encrypted Malicious Traffic Detection." Security and Communication Networks 2023.1 (2023): 7117863.https://doi.org/10.1155/2023/7117863.
27. Zhu, Hui-Juan, et al. "A hybrid deep network framework for android malware detection." IEEE Transactions on Knowledge and Data Engineering 34.12 (2021): 5558-5570. https://doi.org/10.1109/TKDE.2021.3067658.
28. Feng, Ruitao, et al. "A performance-sensitive malware detection system using deep learning on mobile devices." IEEE Transactions on Information Forensics and Security 16 (2020): 1563-1578.https://doi.org/10.1109/TIFS.2020.3025436
29. Dong, Shi, Longhui Shu, and Shan Nie. "Android Malware Detection Method Based on CNN and DNN Bybrid Mechanism." IEEE Transactions on Industrial Informatics (2024). https://doi.org/10.1109/TII.2024.3363016.
30. Shu, Longhui, et al. "Android malware detection methods based on convolutional neural network: A survey." IEEE Transactions on Emerging Topics in Computational Intelligence 7.5 (2023): 1330-1350.https://doi.org/10.1109/TETCI.2023.3281833.
31. Kim, Chiho, et al. "Automated, reliable zero-day malware detection based on autoencoding architecture." IEEE Transactions on Network and Service Management 20.3 (2023): 3900-3914.https://doi.org/10.1109/TNSM.2023.3251282.
32. Fang, Wenbo, et al. "Comprehensive android malware detection based on federated learning architecture." IEEE Transactions on Information Forensics and Security 18 (2023): 3977-3990.https://doi.org/10.1109/TIFS.2023.3287395.
33. Li, Deqiang, et al. "Pad: Towards principled adversarial malware detection against evasion attacks." IEEE Transactions on Dependable and Secure Computing 21.2 (2023): 920-936.https://doi.org/10.1109/TDSC.2023.3265665.

34. Habib, Faiza, et al. "Deep Neural Networks for Enhanced Security: Detecting Metamorphic Malware in IoT Devices." IEEE Access (2024). https://doi.org/10.1109/ACCESS.2024.3383831.
35. Redhu, Ananya, et al. "Deep learning-powered malware detection in cyberspace: a contemporary review." Frontiers in Physics 12 (2024): 1349463. https://doi.org/10.3389/fphy.2024.1349463.
36. Kumar, Manish. "Scalable malware detection system using distributed deep learning." Cybernetics and Systems 54.5 (2023): 619-647.https://doi.org/10.1080/01969722.2022.2068226.
37. Schmitt, Marc. "Securing the Digital World: Protecting smart infrastructures and digital industries with Artificial Intelligence (AI)-enabled malware and intrusion detection." Journal of Industrial Information Integration 36 (2023): 100520.https://doi.org/10.1016/j.jii.2023.100520.
38. Nobakht, Mehrnoosh, Reza Javidan, and Alireza Pourebrahimi. "DEMD-IoT: a deep ensemble model for IoT malware detection using CNNs and network traffic." Evolving Systems 14.3 (2023): 461-477. https://doi.org/10.1007/s12530-022-09471-z.
39. Bhardwaj, Sonam, and Mayank Dave. "Enhanced neural network-based attack investigation framework for network forensics: Identification, detection, and analysis of the attack." Computers & Security 135 (2023): 103521. https://doi.org/10.1016/j.cose.2023.103521.
40. Qin, Maoyang, et al. "A few-shot malware traffic classification method using knowledge transfer learning." 2023 IEEE 23rd International Conference on Communication Technology (ICCT). IEEE, 2023.https://doi.org/10.1109/ICCT59356.2023.10419740.
41. Fang, Wenbo, et al. "Comprehensive android malware detection based on federated learning architecture." IEEE Transactions on Information Forensics and Security 18 (2023): 3977-3990. https://doi.org/10.1109/TIFS.2023.3287395.
42. Amer, Eslam, et al. "Malware detection approach based on the swarm-based behavioural analysis over api calling sequence." 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC). IEEE, 2022. https://doi.org/10.1109/MIUCC55081.2022.9781711

43. Pan, Zhixin, Jennifer Sheldon, and Prabhat Mishra. "Hardware-assisted malware detection and localization using explainable machine learning." IEEE Transactions on Computers 71.12 (2022): 3308-3321. https://doi.org/10.1109/TC.2022.3150573
44. Hashmi, S. A. (n.d.). International Journal of INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING Malware Detection and Classification on Different Dataset by Hybridization of CNN and Machine Learning. In Original Research Paper International Journal of Intelligent Systems and Applications in Engineering IJISAE (Vol. 2024, Issue 6s).